UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/783,246	02/20/2004	Keisuke Inoue	SCEI 3.0-186	1372

530 7590 11/26/2008

LERNER, DAVID, LITTENBERG,
KRUMHOLZ & MENTLIK
600 SOUTH AVENUE WEST
WESTFIELD, NJ 07090

EXAMINER	
KAWSAR, ABDULLAH AL	

ART UNIT	PAPER NUMBER
2195	

MAIL DATE	DELIVERY MODE
11/26/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/783,246	INOUE ET AL.	
	Examiner	Art Unit	
	ABDULLAH AL KAWSAR	2195	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☐ Responsive to communication(s) filed on *02 September 2008*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,3,4,6,7,9-26 and 28-50* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,3,4,6,7,9-26 and 28-50* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *20 February 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *5/23/2008,8/28/2008*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1, 3-4, 6-7, 9-26, 28-50 are pending.

Claim Rejections - 35 USC § 112

2. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

3. Claims 1, 3-4, 6-7, 9-26, 28-50 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

a. The following claim languages are not clearly understood and indefinite:

i. Claim 1, line 11 recites "copying a task table" it is unclear if the entire task table is being copied or the portion of the task table that is related to the processing unit is being copied (i.e. task with same priority or affinity?). Line 15 recites "copying the task table" from "local memory" to "shared memory" it is unclear why the task table is being copied back from to the shared memory and is the entire task table?(i.e. task yield or failure or preempt by higher priority or out of affinity?)

ii. Claims 4, 26 have similar deficiencies as claim 1 above.

iii. Claim 7, lines 10 recite "the list" it is unclear which list is being used to execute that task (i.e. the task queue list or task table list?).

iv. Claims 20, 32 and 45 has similar deficiencies as claim 7 above.

v. Claim 11, line 5 recites "copying task queue and table" it is unclear if the entire task table and queue is being copied or only the entry related to the processor are being copied (i.e. i.e. task with same priority ? affinity?). Lines 10-12 the processor task is copied to the local memory which was already copied to the local memory in lines 5-7, it is unclear why the tasks are being copied twice.

vi. Claims 21, 36 and 46 has similar deficiencies as claim 11 above.

vii. Claim 18, line 3 recites "copied from the local memory to shared memory" it is unclear why the task table is being copied from shared memory to processing unit local memory if the task use is complete.

viii. Claim 19, line 20 recites "modifying the task table entry" it is unclear how the task table is being modified and what is defined by modifying the task table entry (i.e. removing the task from the table? reallocating the task?).

ix. Claims 43 has similar limitations as of claim 19 above.

x. Claim 33, line 2 recites "copied from the shared memory" it is unclear where it is being copied to from the shared memory.

Claim Rejections - 35 USC § 102

4. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

5. Claims 29-31 are rejected under 35 U.S.C. 102(b) as being anticipated by Hirayama (Hirayama) US Patent No. 5592671.

6. As per claim 29, Hirayama teaches the invention as claimed including a multi-processor apparatus, comprising:

a plurality of processing units, each processing unit including a local memory in which to execute processor tasks (col 5, lines 1-3; col 6, lines 9-14); and

a shared memory operable to store: (i) processor tasks that are ready to be executed, and (ii) a task table including a task table entry associated with each of the processor tasks, wherein the processing units are operable to use the task table to determine which of the processor tasks should be copied from the shared memory into their local memories and executed (col 2, lines 54-62; lines 64-67; col 5, lines 33-37; col 6, lines 9-16; lines 37-40).

7. As per claim 30, Hirayama teaches at least some of the task table entries are linked together to achieve at least one list of processor tasks to be invoked in hierarchical order (figure 1; figure 3; col 4, lines 27-32); and

the processing units are operable to use the linked list to determine which of the processor tasks should be copied from the shared memory and executed (col 4, lines 53-58).

8. As per claim 31, Hirayama teaches each of the task table entries includes at least one of (col 2, lines 58-61):

(i) an indication as to whether the associated processor task is ready to be executed by one or more of the processing units (col 4, lines 43-47);

(ii) an indication as to a priority level of the associated processor task (col 3, lines 18-22);

(iii) a pointer to a previous task table entry in a list of task table entries (a previous pointer) (figure 1; col 2, lines 53-67 through col 3, lines 1-17); and

(iv) a pointer to a next task table entry in the list of task table entries (figure 3).

Claim Rejections - 35 USC § 103

9. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

10. Claims 1, 3-4, 6-7, 9-14, 16-22, 26, 28, 32-39 and 41-47 rejected under 35 U.S.C. 103(a) as being unpatentable over Hirayama(Hirayama) US Patent No. 5592671, in view of Bahr (Bahr) EP 0459931.

11. As per claim 1, Hirayama teaches the invention substantially as claimed including a method of managing processor tasks in a multi-processor computing system(abstract, lines 1-3), comprising:

storing the processor tasks in a shared memory that is accessible by a plurality of processing units of the multi-processor computing system (col 2, lines 54-62);

permitting the processing units to determine which of the processor tasks should be executed based on priorities of the processor tasks, wherein the processor tasks that are executed are copied from the shared memory to the local memory of a processing unit (col 2, lines 64-67 through col 3, lines 1-7; col 5, lines 1-42).

Hirayama does not specifically disclose copying a task table from the shared memory to the local memory of a processing unit, the task table entry associated with each of the processor tasks; and copying the task table from the local memory of a processing unit to the shared memory.

However Bahr teaches copying a task table from the shared memory to the local memory of a processing unit, the task table entry associated with each of the processor tasks (col 6, lines 35-43; lines 27-31; col 9, lines 6-10; col 3, lines 20-24; fig. 1); and

copying the task table from the local memory of a processing unit to the shared memory(col 6, lines 31-34).

12. It would have been obvious to a person of ordinary skill in art at the time of invention was made to incorporate the teaching of Bahr into the method of Hirayama to copy the task table from a shared memory to a local memory of a processing unit. The modification would have been obvious because one of the ordinary skills of the art would have utilized the teaching of Bahr to copy the task table form shared memory to local memory for better task execution.

13. As per claim 3, Hirayama teaches the processing units comprise a main processing unit and a plurality of sub-processing units and the sub-processing units access the processor tasks in the shared memory (figure 1).

14. As per claim 4, Hirayama teaches the invention substantially as claimed including a method of managing processor tasks in a multi-processor computing system(abstract, lines 1-3), comprising:

storing the processor tasks in a shared memory that is accessible by a plurality of processing units of the multi-processor computing system(col 2, lines 54-62);

storing a task table in the shared memory, the task table including a task table entry associated with each of the processor tasks (figure 1; col 2, lines 58-62);

linking at least some of the task table entries together to achieve at least one list of the processor tasks to be invoked in hierarchical order (figure 1; figure 3; col 4, lines 27-32); and

permitting the processing units to use the task table to determine which of the processor tasks should be executed in accordance with the list of processor tasks, wherein the processor tasks that are executed are copied from the shared memory to a local memory of a given one of the processing units (col 2, lines 64-67 through col 3, lines 1-7; col 5, lines 1-42).

Hirayama does not specifically disclose copying the task table from the shared memory to the local memory of the given processing unit; and copying the task table from the local memory of the processing unit to the shared memory.

However Bahr teaches copying the task table from the shared memory to the local memory of the given processing unit (col 6, lines 35-43; lines 27-31; col 9, lines 6-10; col 3, lines 20-24; fig. 1); and

copying the task table from the local memory of the processing unit to the shared memory (col 6, lines 31-34).

15. As per claim 6, Hirayama teaches each of the task table entries includes at least one of (col 2, lines 58-61):

(i) an indication as to whether the associated processor task is ready to be executed by one or more of the processing units (col 4, lines 43-47);

(ii) an indication as to a priority level of the associated processor task (col 3, lines 18-22);

(iii) a pointer to a previous task table entry in a list of task table entries (a previous pointer) (figure 1; col 2, lines 53-67 through col 3, lines 1-17); and

(iv) a pointer to a next task table entry in the list of task table entries (figure 3).

16. As per claim 7, Bahr teaches storing a task queue in the shared memory, the task queue including at least one of a head pointer and a tail pointer, the head pointer providing an indication of a first one of the processor tasks in the list, and the tail pointer providing an indication of a last one of the processor tasks in the list (col 8, lines 26-30; lines 14-16; lines 52-53); and

permitting the processing units to use the task table and the task queue to determine which of the processor tasks should be executed in accordance with the list of processor tasks (col 8, lines 4-16).

17. As per claim 9, Bahr teaches linking respective groups of the task table entries together to produce respective lists of processor tasks, each list being in hierarchical order (col 8, lines 26-30); and

providing that the task queue includes respective task queue entries, each entry including at least one of a head pointer and a tail pointer for each of the lists of processor tasks (col 8, lines 52-58 through col 9, lines 1-6).

18. As per claim 10, Hirayama teaches each of the respective lists are associated with processor tasks of a common priority level (col 3, lines 27-30); and

the task queue includes a task queue entry for each of a plurality of priority levels of the processor tasks (col 3, lines 44-50).

19. As per claim 11, Bahr teaches the plurality of processing units includes a plurality of sub-processing units, each of the sub-processing units having local memory(abstract, lines 1-4), further comprising:

copying the task queue and the task table from the shared memory into the local memory of a given one of the sub-processing units (col 6, lines 35-43; col 9, lines 6-10; col 7, lines 23-27);

searching the task queue for the head pointer to a given one of the processor tasks that is ready to be invoked (col 8, lines 11-16); and

copying the given processor task from the shared memory to the local memory of the given sub-processing unit for execution (col 8, lines 4-8; col 9, lines 53-57; col 6, lines 50-54).

20. As per claim 12, Bahr teaches the step of searching the task queue includes searching for the head pointer to a highest priority level one of the processor tasks that is ready to be invoked (col 8, lines 4-8; lines 11-16).

21. As per claim 13, Bahr teaches removing the given processor task from the list of processor task (col 1, lines 24-28).

22. As per claim 14, Bahr teaches each of the task table entries includes a pointer to a next task table entry (col 8, lines 53-56); and

the removal step includes using the next pointer of the given task table entry to change the head pointer to identify the new first processor task as being ready to be next invoked (col 8, lines 53-56; lines 14-16).

23. As per claim 16, Bahr teaches each of the task table entries includes an indication as to whether the associated processor task is READY to be executed or is RUNNING on one or more of the sub-processing units (col 1, lines 34-43); and

the method further includes modifying the given task table entry to indicate that the given processor task is RUNNING (col 1, lines 43-47).

24. As per claim 17, Bahr teaches copying the task queue and the task table from the local memory of the given sub-processing unit into the shared memory when the given sub-processing unit has completed its use thereof (col 6, lines 20-34; col 10, lines 41-49).

25. As per claim 18, Bahr teaches permitting the task queue and the task table to be copied from the shared memory to the local memory of the given sub-processing unit when the given sub-processing unit has completed its use thereof (col 10, lines 41-49).

26. As per claim 19, Hirayama teaches the invention substantially as claimed including a method of managing processor tasks in a multi-processor computing system (abstract, lines 1-3), comprising:

storing the processor tasks in a shared memory that is accessible by a plurality of processing units of the multi-processor computing system(col 2, lines 54-62);

storing a task table in the shared memory, the task table including a task table entry associated with each of the processor tasks (figure 1; col 2, lines 58-62);

linking at least some of the task table entries together to achieve at least one list of processor tasks in hierarchical order (figure 1; figure 3; col 4, lines 27-32);

determining which of the other processor tasks should be executed next within the given processing unit by permitting the given processing unit to use the task table to make such determination(col 2, lines 64-67).

Hirayama does not specifically disclose at least initiating execution a first one of the processor tasks of the list within a given one of the processing units, wherein the first one of the processor tasks yields the given processing unit such that it is capable of executing another of the processor tasks; modifying the task table entry for the first one of the processor tasks that yields the given processing unit; and writing the task table back from the local memory of the given processing unit to the shared memory.

However Bahr teaches at least initiating execution a first one of the processor tasks of the list within a given one of the processing units, wherein the first one of the processor tasks yields the given processing unit such that it is capable of executing another of the processor tasks (col 1, lines 24-28);

modifying the task table entry for the first one of the processor tasks that yields the given processing unit (col 4, lines 20-36); and

writing the task table back from the local memory of the given processing unit to the shared memory (col 6, lines 25-34).

27. As per claim 20, Bahr teaches storing a task queue in the shared memory, the task queue including at least one of a head pointer and a tail pointer, the head pointer providing an indication of a new first one of the processor tasks in the list of processor tasks, and the tail

pointer providing an indication of a last one of the processor tasks in the list of the processor tasks (col 7, lines 2-27; col 8 lines 14-16; lines 53-58 through col 9, lines 1-3); and

permitting the processing units to use the task table and the task queue to determine which of the processor tasks should be executed next (col 8, lines 4-8; lines 14-16).

28. As per claim 21, Bahr teaches the plurality of processing units include a plurality of sub-processing units, each of the sub-processing units having local memory, and wherein the step of determining includes(abstract, lines 1-4):

copying the task queue and the task table from the shared memory into a local memory of the given sub-processing unit (col 6, lines 35-43; col 9, lines 6-10col 7, lines 23-27); and

searching the task queue for the head pointer to the new first processor task that is ready to be invoked (col 8, lines 11-16).

29. As per claim 22, Bahr teaches adding the first processor task back into the list (col 9, lines 13-20).

30. As per claims 26 and 28, they have similar limitations as of claims 1 and 3 above. Therefore they are rejected under the same rational as of claims 1 and 3 above.

31. As per claim 32, 34-39 and 41-42, they have similar limitations as of claims 7, 9-14 and 16-17 above. Therefore they are rejected under the same rational as of claims 7, 9-14 and 16-17 above.

32. As per claim 33, Hirayama teaches the processor tasks that are executed are copied from the shared memory (col 5, lines 33-42).

33. As per claims 43 and 45-47, they have similar limitations as of claims 19 and 20-22 above. Therefore they are rejected under the same rational as of claims 19 and 20-22 above.

34. As per claim 44, Hirayama teaches at least some of the task table entries are linked together to achieve at least one list of processor tasks in hierarchical order (figure 1; figure 3; col 4, lines 27-32).

35. Claims 15, 23-25, 40, 48-50 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hirayama(Hirayama) US Patent No. 5592671, in view of Bahr (Bahr) EP 0459931, in view of Arnon et al. (Arnon) US Patent No. 6321308.

36. As per claim 15, Hirayama and Bahr do not specifically disclose each of the task table entries includes a pointer to a previous task table entry; and the method further includes modifying the previous pointer of the last task table entry of the list to point to the task table entry associated with the new first processor task of the list.

37. However, Arnon teaches each of the task table entries includes a pointer to a previous task table entry (col 4, lines 49-54); and

the method further includes modifying the previous pointer of the last task table entry of the list to point to the task table entry associated with the new first processor task of the list (col 7, lines 52-62).

38. It would have been obvious to a person of ordinary skill in art at the time of invention was made to incorporate the teaching of Arnon into the combined method of Hirayama and Bahr to have a previous pointer to point to the next task table entry. The modification would have been obvious because one of the ordinary skills of the art would have a previous and next pointer on a linked list based task table to be able to modify the pointers for add and delete operations.

39. A per claim 23, Arnon teaches each of the task table entries includes a pointer to a next task table entry and pointer to a previous task table entry (col 4, lines 49-58); and

the step of adding includes modifying the linking of the task table entries to include links to the task table entry associated with the first processor task (col 7, lines 3-7; lines 26-35).

40. As per claim 24, Arnon teaches wherein the step of modifying the linking of the task table entries includes linking the task table entry associated with the first processor task between a prior task table entry and a following task table entry that were previously linked to one another (col 7, lines 24-32).

41. As per claim 25, Arnon teaches modifying the next pointer of the prior task table entry to point to the task table associated with the first processor task (col 7, lines 39-41);

modifying the previous pointer of the task table entry associated with the first processor task to point to the prior task table entry (col 7, lines 38-39);

modifying the next pointer of the task table entry associated with the first processor task to point to the following task table entry (col 7, lines 42-45); and

modifying the previous pointer of the following task table entry to point to the task table entry associated with the first processor task (col 7, lines 45-49).

42. As per claim 40, it has similar limitations as of claim 15 above. Therefore it is rejected under the same rational as of claim 15 above.

43. As per claims 48-50, they have similar limitations as of claims 23-25 above. Therefore they are rejected under the same rational as of claims 23-25 above.

Response to Arguments

44. Applicant's arguments filed 09/02/2008 in regards to claims 1, 4, 19, 26 and 43 have been fully considered but they are moot in view of new ground of rejection. Applicant's argument in regards to claim 29 has been fully considered but not persuasive.

45. In the remarks applicant argues:

(1) Hirayama fails to teach "determine which of the processor task should be copied from the shared memory into their local memories and executed".

46. Examiner respectfully disagree with applicant:

i. As to point (1), applicant supports his argument mentioning that the cited portion of Hirayama teaches general block diagram of the resource management system. process management table, the entries, run queue and how the system prioritizes and allocates the pages of the processor's local memory. Examiner respectfully disagrees with the applicant. The claimed limitations if brad and does not specify what is the criteria to determine which task should be copied and moreover in line 4 and 8 claim recites "operable" which is open ended thus indefinite and does not mandate the use of shared memory or task table to determine or copy task tasks from shared to local memory. Hirayama teaches determining tasks to be executed based on priority and the tasks are being assigned to the processor based on priority. Hirayama also teaches prioritizing and allocating memory pages of the processor's local memory wherein the pages are allocated to process(task) for execution. Since the task of the highest priority is assigned for execution the memory table of the highest priority will be assigned to the task which is the processor local memory (col 2, lines 50-67 through col 3, lines 1-16; col 5, lines 1-42; col 6, lines 37-40).

Conclusion

47. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

48. A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

49. Any inquiry concerning this communication or earlier communications from the examiner should be directed to ABDULLAH AL KAWSAR whose telephone number is (571)270-3169. The examiner can normally be reached on 7:30am to 5:00pm, EST.

50. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng Ai T. An can be reached on 571-272-3756. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

51. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Meng-Ai An/
Supervisory Patent Examiner, Art Unit 2195

/Abdullah-Al Kawsar/
Examiner, Art Unit 2195